First order primal-dual algorithms for total variation methods with applications to image analysis

Dong Nie, April 27th

Abstract: Total variation is introduced in previous classes, and is widely used in image analysis, such as image denoising problem, image segmentation problems and so on. Total variation minimization is very important in convex variational methods for image analysis (it allows sharp discontinuities in solutions). While total variation is non-smooth, which limits us to minimize the variational methods with total variation regularization. Chambolle *et al.* [1] proposed a first order primal-dual algorithms for non-smooth convex optimization problems (including total variation) in image analysis. The proposed approach provides good solutions for the total variation based algorithms. The conducted experiments demonstrate that the primal-dual algorithms are faster compared to traditional widely used methods (e.g., gradient descent); furthermore, primal-dual algorithms can solve problems which primal or lagragian-dual approaches are hard to solve.

1. Introduction

In image analysis fields, we usually have to determine the model parameters based on observations. And there are obviously inverse problems. Researchers often formulate the problems as Eq. 1.

$$\min_{x \in X} F(x) + G(x) \quad (1)$$

where F is loss function and G is regularization term. And Eq. 1 tries to find a tradeoff between data fit and smoothness. Optimization algorithms are developed to solve such problems. If the objective function is convex (and the domain is also a convex set), then we are theoretically guaranteed to have a global optimal.

When it comes to tasks in image analysis, images exhibit large extent of spatial coherence. Given the intensity of some pixel, it is very likely that its neighboring pixels have the same intensity. And total variation (TV), shown in Eq. 2, is a good candidate for image analysis problems which allows for sharp discontinuities in the solution.

$$\mathrm{R}(u) = \int_{\Omega} |\nabla u| dx \quad (2)$$

In recent years, TV-based models have been successfully used to solve various image analysis problems. However, as TV is non-smooth, we can not easily compute the gradient directly. Unfortunately, A lot of optimization literature focus on problems where either F or G are smooth, but they cannot be applied to where both F and G are non-smooth.

Several algorithms have been developed to solve the non-smooth convex optimization (of course, these algorithms also suits for smooth convex problems). Projected (sub)gradient descent [4] is a conventional tool for the L^1 norm problems (a classical non-smooth convex problems), while it may be blocked by a low accuracy issue. Vogel *et al.* proposed to solve the same Euler-Lagrange equation of (2) via a fixed point iteration method [6]. Alternating direction method of multipliers (ADMM) [5] is an algorithm that separate the convex optimization problems into smaller ones which are easier to solve, and ADMM is traditionally considered to be a fast method of L^1 related problems.

Recently, Chambolle *et al* [1] proposes a first-order primal-dual algorithm for convex problems, which is especially a good approach for non-smooth convex problems. Chambolle's algorithm focus on a class of problems described in Eq. 3.

$$\min_{x \in X} F(Kx) + G(x) \quad (3)$$

where K is a linear operator, $G: X \to [0, +\infty)$ and $G: X \to [0, +\infty)$. In this paper, we only consider that F and G are both convex functions. Note, X is a finite-dimensional real vector space. Eq. 3 is the general problem we will discuss in this paper.

To introduce the primal-dual and dual formation of Eq. 3, we have to know Legendre-Fenchel conjugate, shown in Eq. 4

$$F^*(p) = \sup_{x \in X} \langle p, x \rangle - F(x) \quad (4)$$

Now we introduce a dual variable y.Then we can write the primal-dual and dual formulation in Eq. 5 and Eq. 6.

$$\min_{x \in X} \max_{y \in Y} \langle Kx, y \rangle + G(x) - F^*(y) \quad (5)$$

$$\max_{y \in Y} -\big(G^*(-K^*y) + F^*(y)\big) \quad (6)$$

We have provided proof for transformation from primal form to primal-dual and dual form according to my own knowledge in appendix I. You can also refer to [1] for more details about the formulations (in fact, there is no direct proof in [1]). In later sections, we will go deep into this new convex optimization tools.

2. Algorithm

The big picture of this first-order primal-dual algorithm is that: first transform primal form to primal-dual form and then alternate proximal steps that alternately maximize and minimize a primal-dual form of the saddle function. In fact, after being transferred into primal-dual formulation, the non-smooth term is removed or can be solved with the help of proximity operator. The big idea can be simply described in Fig. 1.

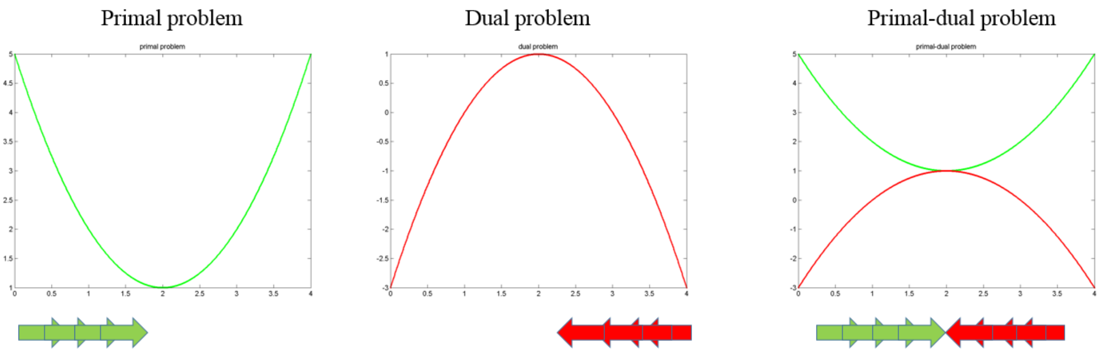

Fig. 1 Simple explanation of how primal-dual algorithm works

The primal-dual formulation we consider in this paper is shown in Eq. 5. We can simply separate Eq. 5 into two objective functions as Eq. 6 which are with respect with primal variable x and dual variable y, respectively.

$$\begin{cases} \max\limits_{y \in Y} \langle Kx, y \rangle - F^*(y) - \frac{\lambda_1}{2} \|y - y^n\|_2^2 \to Q(y) \\ \min\limits_{x \in X} \langle Kx, y \rangle + G(x) + \frac{\lambda_2}{2} \|x - x^n\|_2^2 \to P(x) \end{cases} \quad (6)$$

Solving Eq. 6, we can achieve explicit solutions for x and y, show.

$$\begin{cases} (\nabla Q)_y = 0 \Rightarrow y = (I + \partial F^*)^{-1}(y^n + \sigma K x) \\ (\nabla P)_x = 0 \Rightarrow x = (I + \tau \partial G)^{-1}(x^n - \tau K^* y) \end{cases} \quad (7)$$

Now we can form an algorithm to iteratively update the primal and dual variable. The steps are summarized in Algorithm 1.

Algorithm 1 (PDCP1):

- Initialization: choose $\tau, \sigma > 0, \theta \in [0,1], (x^0, y^0) \in X \times Y$ and set $\bar{x}^0 = x^0$
- Iterations $(n \geq 0)$: Update x^n, y^n, $\bar{x}^n$ as follows:

$$\begin{cases} y^{n+1} = (I + \sigma \partial F^*)^{-1}(y^n + \sigma K \bar{x}^n) \\ x^{n+1} = (I + \tau \partial G)^{-1}(x^n - \tau K^* y^{n+1}) \\ \bar{x}^{n+1} = x^{n+1} + \theta(x^{n+1} - x^n) \end{cases}$$

The convergence rate of this algorithm is $O(n)$. For the convergence details, you can once again refer to [1].

If the two subproblems (I mean F^* and G) are uniformly convex, we can achieve $O(N^2)$ convergence rate. For the convergence rate details, you can refer to [4, 7]. The algorithm is summarized as Algorithm 2.

Algorithm 2 (PDCP2):

- Initialization: choose $\tau_0, \sigma_0 > 0$ with $\tau_0\sigma_0 L^2 \leq 1$, $\theta \in [0,1], (x^0, y^0) \in X \times Y$ and set $\bar{x}^0 = x^0$
- Iterations ($n \geq 0$): Update x^n, y^n, $\bar{x}^n$, θ_n, τ_n, σ_n as follows:

$$\begin{cases} y^{n+1} = (I + \sigma_n \partial F^*)^{-1}(y^n + \sigma_n K\bar{x}^n) \\ x^{n+1} = (I + \tau_n \partial G)^{-1}(x^n - \tau_n K^* y^{n+1}) \\ \theta_n = 1/\sqrt{1+2\gamma\tau_n}, \tau_{n+1} = \theta_n\tau_n, \sigma_{n+1} = \sigma_n/\theta_n \\ \bar{x}^{n+1} = x^{n+1} + \theta_n(x^{n+1} - x^n) \end{cases}$$

3. Application on ROF model

We now apply the first order primal-dual algorithm on the famous "Rudin-Osher-Fatemi" model [2] which is widely used in image denoising.

$$\min_x \int_\Omega |Du| + \frac{\gamma}{2}\|u-g\|_2^2 \quad (8)$$

where $\Omega \subset R^d$ is the image domain, $g \in L^1(\Omega)$ is the noisy input image, $g \in L^1(\Omega)$ is the denoised output image and γ is the parameter to tradeoff between data fitting term and regularization term. $\int_\Omega |Du|$ is the total variation of u where Du denotes the distributional derivative, and this helps preserve sharp edges in the image which is quite desired in most image analysis problems.
The discrete version of Eq. 8 is given in Eq. 9.

$$h^2 \min_{u\in X}\|\nabla u\|_1 + \frac{\gamma}{2}\|u-g\|_2^2 \quad (9)$$

where $\|\nabla u\|_1 = \sum_{i,j}|(\nabla u)_{ij}|$, and i, j indexes location in the image.

3.1 Primal-dual formulation

In the case of ROF model, the $F(Kx)$ in Eq. 5 is $\|\nabla u\|_1$, thus, we can inference that

$$F^*(p) = \delta_P(p) = \begin{cases} 0, \ p \in P \\ +\infty, \ p \notin P \end{cases} \quad (10)$$

Then we can rewrite Eq. 9 in a primal-dual form:

$$\min_{u\in X}\max_{p\in Y} -\langle u, \operatorname{div} p\rangle_X + \frac{\gamma}{2}\|u-g\|_2^2 - \delta_P(p) \quad (11)$$

where p is a dual variable, div is divergence operator, and $P = \{p \in Y: \|p\|_\infty \leq 1\}$
The above is to inference the primal-dual formulation from the view of definition stated in Eq. 5. We can also provide another explanation using simple lagragian multiplier knowledge.

We first give a dual norm [8] for $\|\nabla u\|_1$ in Eq. 12. We can transform the constraint form in Eq. 12 into unconstraint one in Eq. 13. And the divergence operator follows Eq. 14. With these three equations (Eq. 12, Eq. 13 and Eq. 14), we can easily rewrite Eq. 9 in the primal-dual form as Eq. 11.

$$\|\nabla u\|_1 \Leftrightarrow \max_p < \nabla u, p >, \ s.t. \ \|p\|_\infty \leq 1 \quad (12)$$

$$\|\nabla u\|_1 \Leftrightarrow \max_p < \nabla u, p \succ -\delta_P(p) \quad (13)$$

$$\langle \nabla u, p \rangle \Leftrightarrow -\langle u, \mathrm{div} p \rangle \quad (14)$$

3.2 “Proximal” operator

We can compute the primal and dual variables following Eq. 7, which is shown in Eq. 15 and Eq. 16.

$$u = (I + \tau \partial G)^{-1}(\tilde{u}) \quad (15)$$

$$p = (I + \sigma \partial F^*)^{-1}(\tilde{p}) \quad (16)$$

We can solve Eq. 15 and Eq. 16 with the help of proximity operator. The proximity operator of a function can be explained in Eq. 17.

$$(I + \tau \partial F)^{-1}(x) = \underset{y}{\mathrm{argmin}} \frac{1}{2} \|y - x\|^2 + \tau F(y) \quad (17)$$

As $F^*(p) = \delta_P(p)$ in ROF model, so we can get p by Eq. 18.

$$(I + \sigma \partial F^*)^{-1}(\tilde{p}) = proj_P(p) \quad (18)$$

As

$$P = \{p \in Y : \|p\|_\infty \leq 1\} \quad (19)$$

Thus,

$$proj_P(p) = \frac{p}{\max(1, \|p\|)} \quad (20)$$

Similarly, we can compute the primal variable u,

$$(I + \tau \partial G)^{-1}(\tilde{u}) = \frac{u + \gamma \tau g}{1 + \gamma \tau} \quad (21)$$

Having Eq. 20 and Eq. 21, we can finally apply Algorithm 1 and Algorithm 2 on ROF model.

3.3 Experiments

To better demonstrate the advantage of this method, we conduct some experiments with this first-order primal-dual algorithm.

As γ is tradeoff between data fitting and regularization, we first test the proposed algorithm (PDCP1) on different setting of this parameter. And the visualization is shown in Fig. 2.

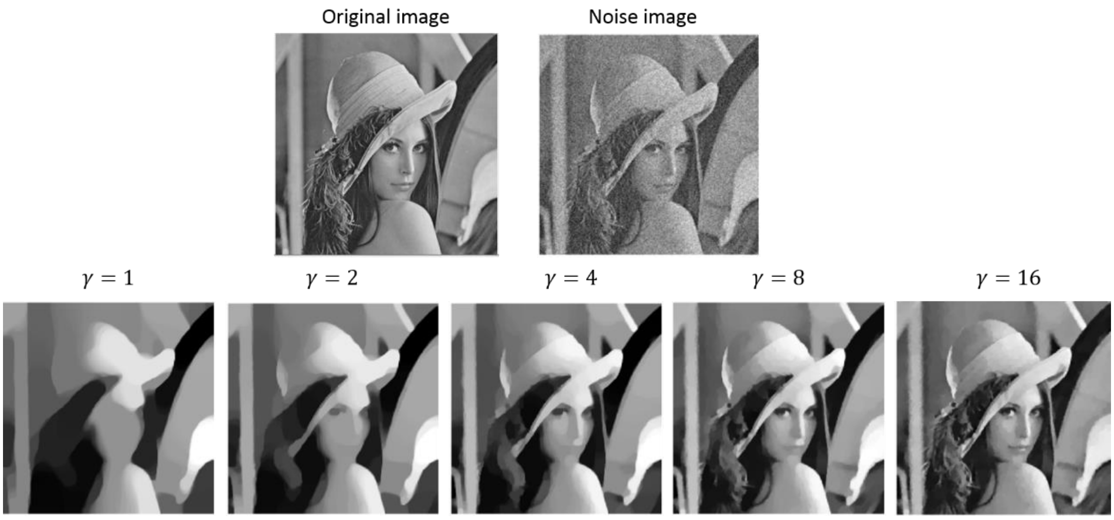

Fig. 2 The denoising algorithm by ROF model solved by the primal-dual algorithm. The first row is original image (without noise), and noise input image. The second row is denoised results achieved by different γ

As shown in Fig. 2, the denoising effect will be best when γ is set at 8. And the sharp edges are preserved in the denoised images. This can be explained by the fact that more flat regions appear in the image for stronger regularization.

Also, we test first-order primal-dual (PDCP1 algorithm) at different noise level. The experimental setting are the same except the input image is at different noise level (The stop condition used here is 400 iterations). The results are presented in Fig. 3.

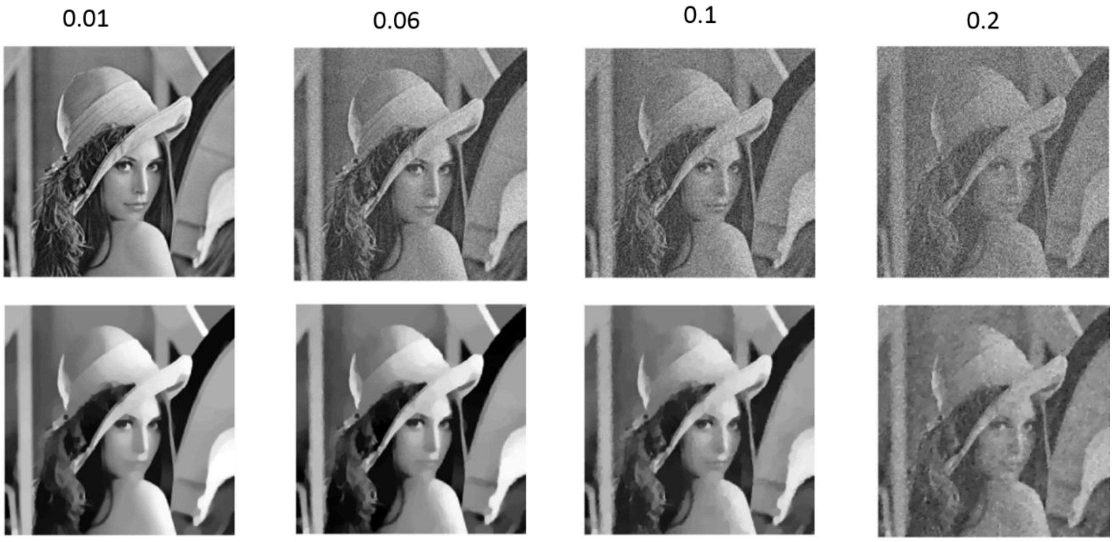

Fig. 3 ROF model (solved by PDCP1) on different noised images. The first row is images at different noise level, and second row is corresponding denoised images

Furthermore, we adopt several commonly used algorithm for solving ROF model as comparison, and we list them as follows:

- Gradient Descent (GD): The $|\cdot|$ is not differentiable at 0, so we can replace $|\cdot|$ by a function which behaves almost like $|x|$. Then we can use gradient descent to compute the primal and dual variables.

$$|x| \approx \frac{x^2}{\sqrt{x^2+\varepsilon^2}} \quad (22)$$

- PDE based: We first introduce a dual variable and rewrite the minimization problem as in Eq. 8. The gradient can be obtained by calculus of variations, and then we can solve the PDE (shown in Eq. 23 and Eq. 24) to get u with iterative methods.

$$p_\tau = \nabla u, \|p\| \leq 1 \quad (23)$$
$$u_\tau = \mathrm{div} p + \gamma(I - U) \quad (24)$$

- ADMM

Alternating directions method of multipliers (ADMM) can be employed to solve Eq. 8, showing in Eq. 25.

$$\min_{u,p,\lambda} \|p\|_1 + \frac{\gamma}{2}\|u-g\|^2 + \langle \lambda, p - \nabla u \rangle + \frac{\alpha}{2}\|p - \nabla u\|^2 \quad (25)$$

We set the same experimental settings again, the stop condition is that the normalized energy (energy of primal problem) reduction to $1e-4$. And the tradeoff parameter γ is set 8 and 16 respectively. We record the cost time, and show in Fig. 4.

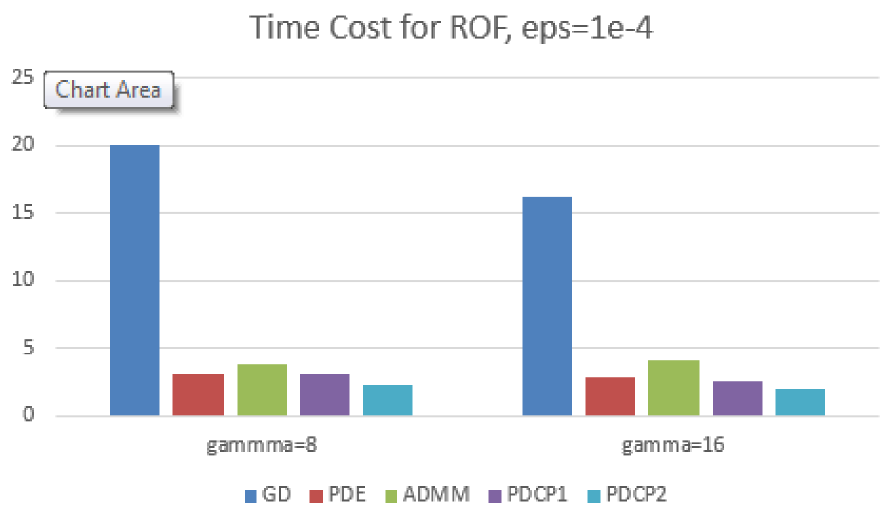

Fig. 4 Time cost comparison for five algorithms solving ROF model

Fig. 4 tells us that PDCP2 methods (acceleration one) gives the best performance (time cost). And PDCP1 also gives better results than ADMM which is conventionally considered a fast algorithm to solve L^1 related problems. In fact, PDCP2 gives $O(N^2)$ convergence rate, PDCP1 and ADMM give $O(N)$ convergence rate.

The above experiments basically assure that the first-order primal-dual algorithms work well on ROF model, and demonstrate that it is an efficient algorithm for solving TV-based models.

4. Application on TV-L1 models

Similar to ROF model, TV-L1 model can be expressed as Eq. 26.

$$\min_{x} \int_{\Omega} |Du| + \gamma \|u - g\|_1 \quad (26)$$

The discretized TV-L1 model is shown in Fig. 27.

$$h^2 \min_{u \in X} \|\nabla u\|_1 + \gamma \|u - g\|_1 \quad (27)$$

Once again, the primal-dual form can be achieved by Eq. 28.

$$\min_{u \in X} \max_{p \in Y} -< u, \operatorname{div} p >_X + \gamma \|u - g\|_1 \; - \delta_P(p) \quad (28)$$

To solve Eq. 28, we have

$$p = (I + \sigma \partial F^*)^{-1}(\tilde{p}) \Leftrightarrow p_{i,j} = \frac{\tilde{p}_{i,j}}{max(1, |\tilde{p}_{i,j}|)} \quad (29)$$

$$u = (I + \tau \partial G)^{-1}(\tilde{u}) \Leftrightarrow u_{i,j} = shrink(u, g, \gamma\tau) \quad (30)$$

where

$$\text{shrink}(u, g, \tau\gamma) = \begin{cases} u - \tau\gamma, u > g + \gamma\tau \\ g, |u - g| \le \gamma\tau \\ u + \tau\gamma, u < g - \gamma\tau \end{cases} \quad (31)$$

We again conduct an experiment with different tradeoff parameter, and the results are shown in Fig. 5.

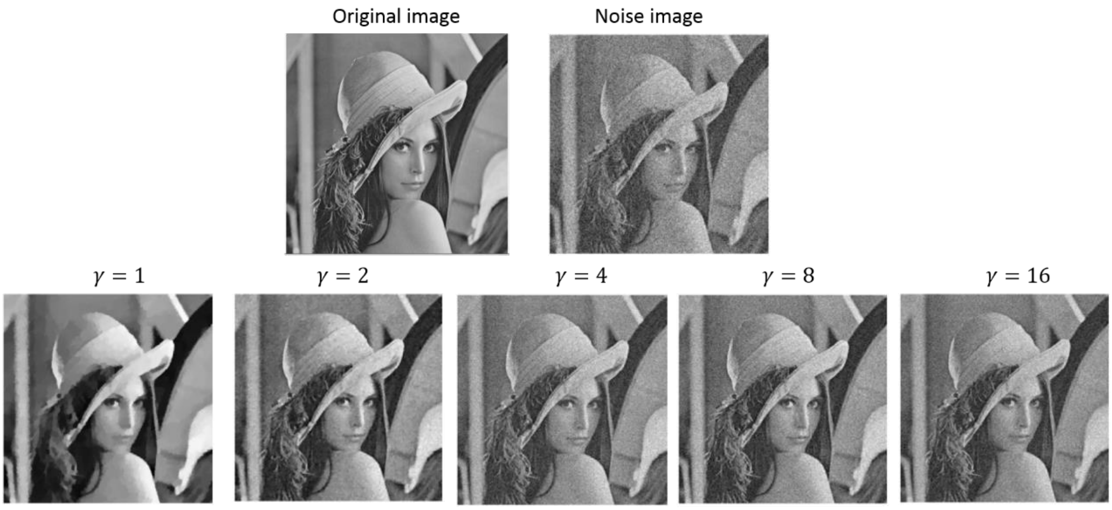

Fig. 5 The denoising algorithm by TVL1 model solved by the primal-dual algorithm. The first row is original image (without noise), and noise input image. The second row is denoised results achieved by different γ

Different from ROF model, the best performance are achieved by $\gamma = 1$. When the γ increases, the performance become worse.

5. Discussion

The first-order primal-dual algorithm is a general framework for convex optimization problems which has the form of Eq. 1. And it can be generalized to more problems.

At first, all total variation based image analysis problems can adopt the first-order primal-dual algorithm to solve, such as image deconvolution, image zooming, image inpainting, image segmentation and so on.

Moreover, the primal-dual algorithm is suitable for many machine learning algorithms, in which data fitting term and regularization term are both convex, such as SVM, group lasso, multi-task learning, matrix factorization and so on.

In general, the first-order primal-dual algorithm is an example of a first order method, meaning it only requires functional and gradient evaluations which are feasible for many problems. Also, it is also an example of a primal-dual method. Each iteration updates both a primal and a dual variable. The first-order primal-dual algorithm stated in this paper is a algorithm framework suitable for problems in which sub-problems are convex, especially non-smooth sub-problems.

Reference

[1]. Chambolle, A., & Pock, T. (2011). A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, *40*(1), 120-145.

[2] Rudin, L. I., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, *60*(1), 259-268.

[3] Zhu, M., & Chan, T. (2008). An efficient primal-dual hybrid gradient algorithm for total variation image restoration. *UCLA CAM Report*, 08-34.

[4] Beck, A., & Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*,*2*(1), 183-202.

[5] Wen, Z., Goldfarb, D., & Yin, W. (2010). Alternating direction augmented Lagrangian methods for semidefinite programming. *Mathematical Programming Computation*, *2*(3-4), 203-230.

[6] Vogel, C. R., & Oman, M. E. (1996). Iterative methods for total variation denoising. *SIAM Journal on Scientific Computing*, *17*(1), 227-238.

[7] Nesterov, Y. (1983, February). A method of solving a convex programming problem with convergence rate O (1/k2). In *Soviet Mathematics Doklady*(Vol. 27, No. 2, pp. 372-376).

[8] https://inst.eecs.berkeley.edu/~ee127a/book/login/def_dual_norm.html